Sketch inference as a theory of visual contour computation

Jonas August

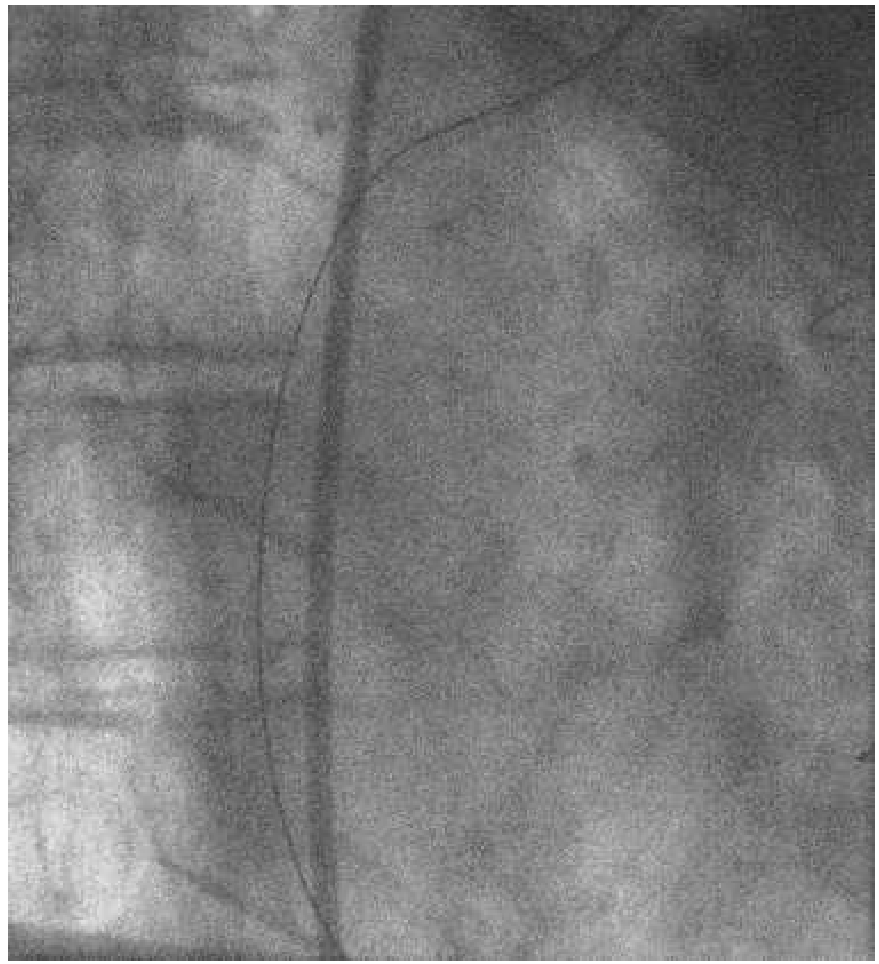
Robotics Institute
Carnegie Mellon University

Acknowledgements

- Steve Zucker (advisor at Yale)
- Nick Hengartner (LANL, Yale)
- Vladimir Rokhlin (Yale)
- David Mumford (Brown)
- Lance Williams (UNM)
- Karvel Thornber (NEC)
- Takeo Kanade (CMU)

Application:
Sketch Enhancement

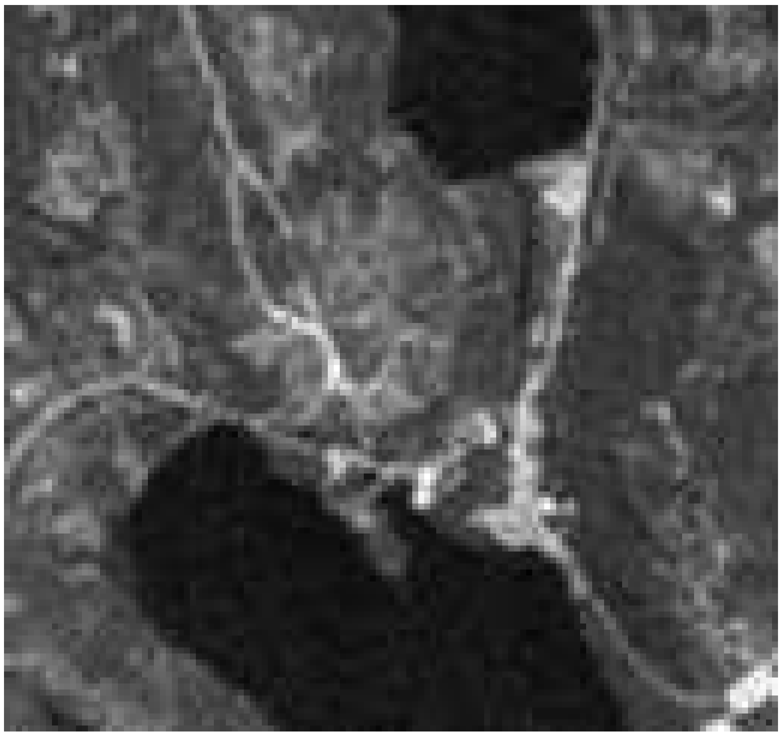
Application: Reducing Fluoroscopic Exposures



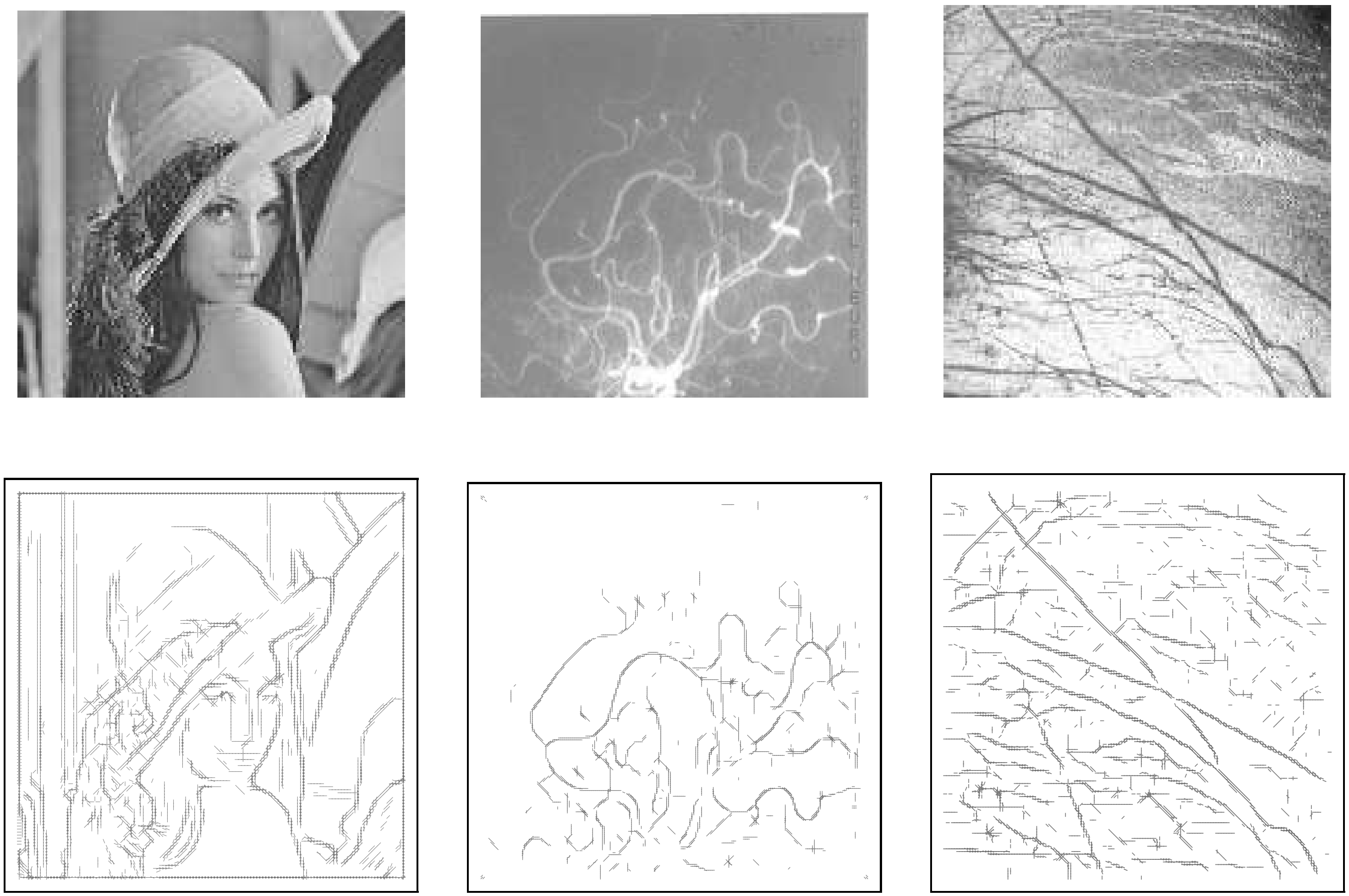
[Viergever et al]

Application:

Finding Roads and Rivers in Satellite Imagery

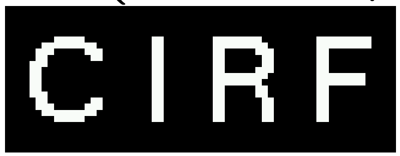


Images with Contours

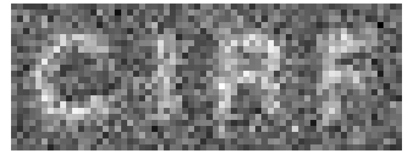


Local edge & line measurements

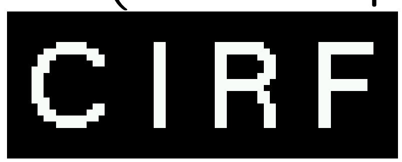
Original (No corruption)



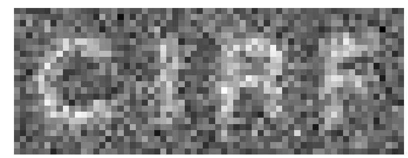
With Blur and Noise



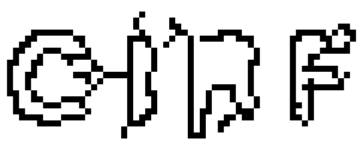
Original (No corruption)



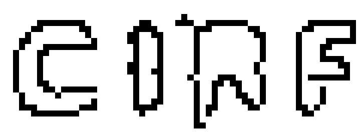
With Blur and Noise



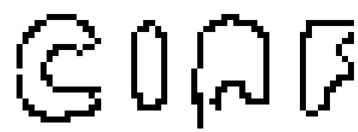
Canny Edges:



$\sigma = 1$



$\sigma = 1.5$



$\sigma = 2$

$\boxed{\text{Goal: Sketch Inference}}$

Fusion of Differential Geometry and Random Fields
by Eliminating Curve Parameterization

$\boxed{\text{Goal: Sketch Inference}}$

Fusion of Differential Geometry and Random Fields by Eliminating Curve Parameterization

Outline

- Background
- Direction Process
- The Curve Indicator Random Field + All Cumulants
- Empirical Edge Statistics

- Curvature Process and Euler Spirals

- Volterra Filters and Partial Differential Equations
  for Enhancing Curve Images

Inference for a Single Contour

- Dynamic programming [Montanari ’71; Sha’ashu & Ullman ’88]

- Heuristic search [Martelli ’76]

- Bayesian [Geman & Jedynak ’96, Yuille & Coughlan ’00]

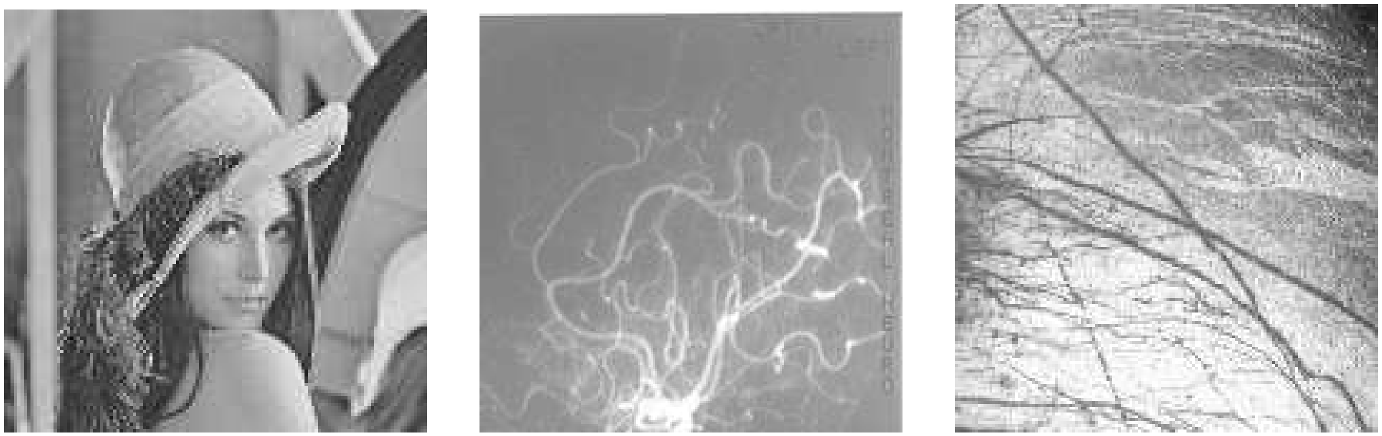
Inference for Multiple Contours

- Local edge detection + linking (non-contextual)

- Context via **local interactions**:
  - MRFs [Geman & Geman; Marroquin]
  - Energy-based [Mumford& Shah, Nitzberg et al, Williams]
  - Dictionary-based relaxation labeling [Hancock et al]
  - Relaxation labeling with co-circularity [Zucker,Parent,Iverson]

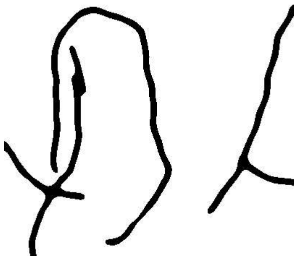
- Explicit parameterizations and MCMC simulation [Zhu et al]

Images with Contours



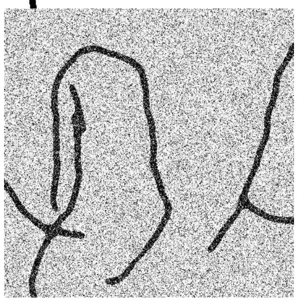
Sketch Realizations

Inferring a Sketch



$U_i =$ underlying random field (ideal sketch)

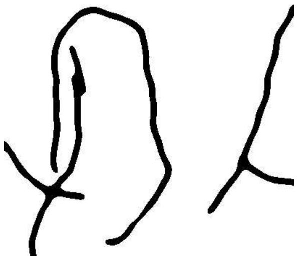
("indicates" curve at $i = (x, y, \theta)$)



$M_i =$ measurement random field

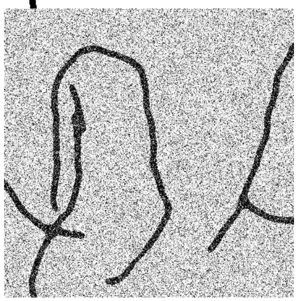
(corrupted form of U_i, i.e., from local edge operator, e.g. image gradient)

Inferring a Sketch



U_i = underlying random field (ideal sketch)

("indicates" curve at $i = (x, y, \theta)$)

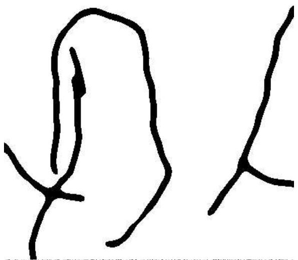


M_i = measurement random field

(corrupted form of U_i, i.e., from local edge operator, e.g. image gradient)

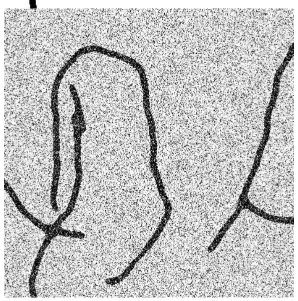
Goal: Estimate U_i given M_i.

Inferring a Sketch



U_i = underlying random field (ideal sketch)

("indicates" curve at $i = (x, y, \theta)$)



M_i = measurement random field

(corrupted form of U_i, i.e., from local edge operator,

e.g. image gradient)

Goal: Estimate U_i given M_i.

Posterior: $\mathbb{P}(U|M) \propto \mathbb{P}(M|U)\mathbb{P}(U)$

Likelihood $\mathbb{P}(M|U)$: corruption model (noise and blur)

$\boxed{\textbf{Prior probability } \mathbb{P}(U)}$ "bias" to overcome uncertainty

What's a prior for sketches?

Filtering

Linear Filters: Model: $M = \text{blur}(U) + \text{noise}$

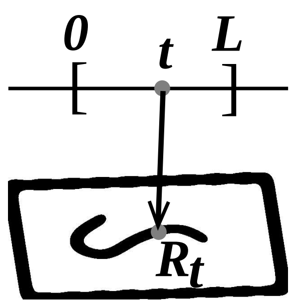
Linear minimum mean square error estimate:
Requires *second moments* of U.

Quadratic and Higher-Order Filters:

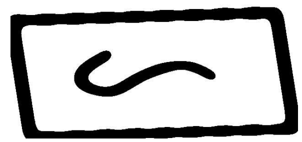
Require *higher moments* of U.

$\boxed{\textit{Where do the moments come from?}}$

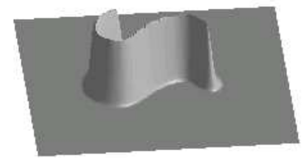
Approach to Sketch Inference



$\longrightarrow$



$\longrightarrow$



$\longrightarrow$ Moments

$\big\downarrow$

Random
parameterization
$t \mapsto R_t$

Random **set**
$\{R_t : t \in [0, L]\}$

Random **indicator**
"Curve indicator
random field" U
(Unknown sketch)

Sketch
Filters

Eliminate curve parameterization by accumulating "ink"

Differential Geometry of Planar Curves

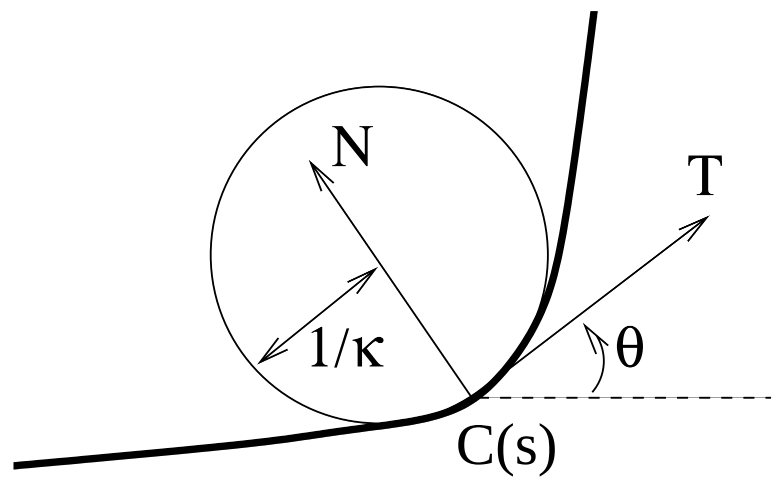
Curve with parameter $s \in \mathbb{R}$: $\quad C : s \mapsto C(s) = (x(s), y(s)) \in \mathbb{R}^2$

Tangent vector $= T = \frac{C'}{||C'||}$,
where $C' = (\frac{dx}{ds}, \frac{dy}{ds})$

Normal vector $= N = \text{rotate}_{90^\circ} T$

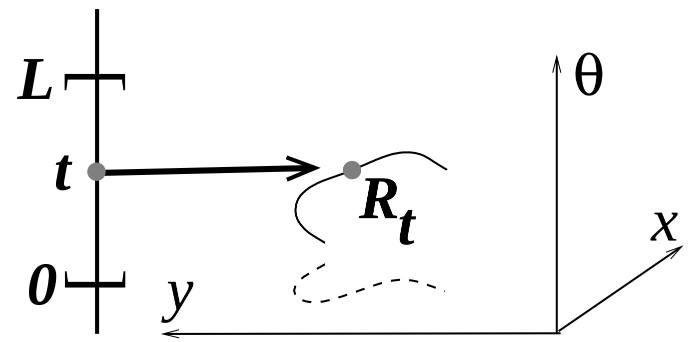
Direction $= \theta$: $T = (\cos\theta, \sin\theta)$

Curvature $\kappa = \frac{d\theta}{ds}$



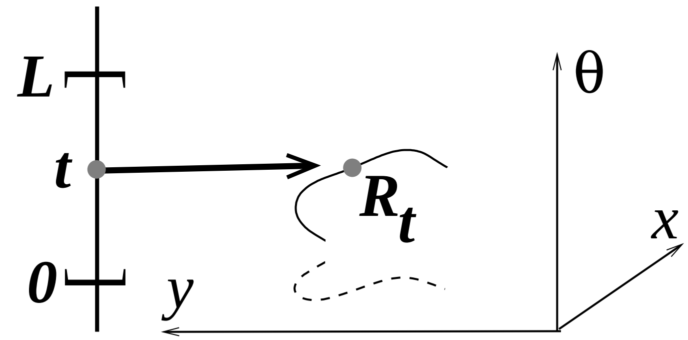
A Markov Process with Direction

Lift of curve: $t \mapsto R_t = (x, y, \theta)$



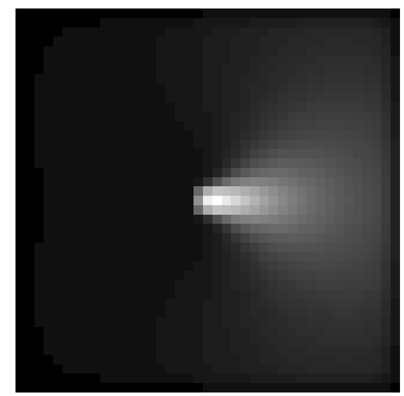
A Markov Process with Direction

Lift of curve: $t \mapsto R_t = (x, y, \theta)$



Mumford's process with *direction*:

$$\dot{x} = \cos\theta \quad \dot{y} = \sin\theta \quad \dot{\theta} = \text{noise}$$



Green's function $G = (g_{ij})$

Developed by Williams, Jacobs, Thornber, Zweck.

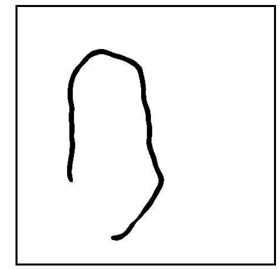
Approximate continuous space discretely: $i = (x, y, \theta)$

$$g_{ij} = \begin{cases} \text{time spent in } j \\ \text{given process} \\ \text{started in } i \end{cases}$$

Curve Indicator Random Field

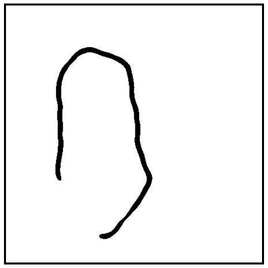
Discrete-space Markov process $R_t = i = (x, y, \theta), \quad t \in [0, L]$
Random length $L \sim$ exponential(α^{-1})

Key intuition: Let $V_i \approx \begin{cases} 1, & \text{if } i \text{ is on the curve} \\ 0, & \text{otherwise} \end{cases}$



Curve Indicator Random Field

Discrete-space Markov process $R_t = i = (x, y, \theta), \quad t \in [0, L]$
Random length $L \sim$ exponential(α^{-1})



Key intuition: Let $V_i \approx \begin{cases} 1, & \text{if } i \text{ is on the curve} \\ 0, & \text{otherwise} \end{cases}$

$$\mathbf{1}\{\text{condition}\} = \begin{cases} 1, & \text{if } \textit{condition} \text{ true} \\ 0, & \text{otherwise} \end{cases}$$

Definition: *Curve indicator random field (1 curve):*

$$\begin{aligned} V_i &:= \int_0^L \mathbf{1}\{R_t = i\} dt \\ &= \text{time spent by curve at position } i = (x, y, \theta) \end{aligned}$$

Curve Interactions

How are crossings represented?

Using parameterization:
- must check all t_1, t_2 whether $R_{t_1} = R_{t_2}$
- global computation

Using CIRF:
- ink buildup occurs at crossings
- local computation: U_i^2

Theoretical Result:

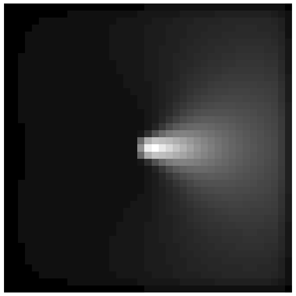
All Joint Moments of the Curve Indicator Random Field

Claim (Single curve case):

Positions $i_1, \ldots, i_k \in \{(x, y, \theta)\}$

$$\mathbb{E}[V_{i_1} \cdots V_{i_k}] \propto \sum g_{j_1 j_2} \cdots g_{j_{k-1} j_k}$$

Sum over permutations $j_1, \cdots, j_k$ of $i_1, \cdots, i_k$



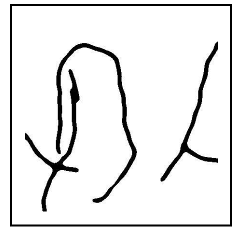
$$g_{ij} = \begin{cases} \text{time spent in } j \\ \text{given process} \\ \text{started in } i \end{cases}$$

Sum over all moments gives Feynman-Kac formula.

A Sketch with Multiple Curves

Random number $\mathcal{N}$ of i.i.d. Markov processes $R_t^{(1)}, \ldots, R_t^{(\mathcal{N})} \sim R_t$
Independent random lengths $L_1, \cdots, L_{\mathcal{N}} \sim L$

Take superposition of i.i.d. 1-curve CIRFs:



Definition: *Curve indicator random field (multiple curves):*

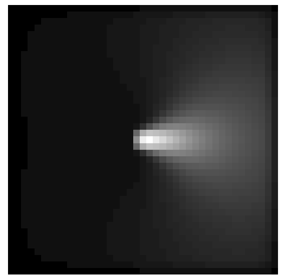
$$U_i := \sum_{n=1}^{\mathcal{N}} \int_0^L \mathbf{1}\{R_t^{(n)} = i\} dt$$

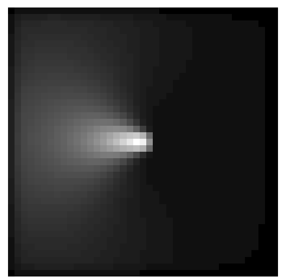
Claim: cumulant$\{U_{i_1}, \ldots, U_{i_k}\} \propto \sum g_{j_1 j_2} \cdots g_{j_{k-1} j_k}$
Sum over permutations $j_1, \cdots, j_k$ of positions $i_1, \cdots, i_k$

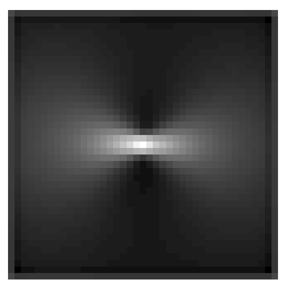
Corollary: The curve indicator random field is non-Gaussian.

Covariance of Curve Indicator Random Field

$$\mathrm{cov}(U_i, U_j) \quad \propto \quad [g_{ij} + g_{ji}]$$

$=$  g_{ij} (forward)

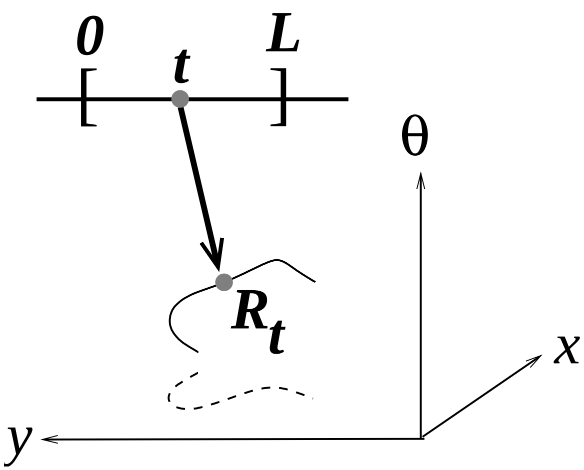
$+$  g_{ji} (backward)

$=$ 

(Integrated over θ for display)

Covariance of Curve Indicator Random Field

$t \mapsto R_t = (x, y, \theta)$



"Ideal" edge correlations with horizontal edge at center:

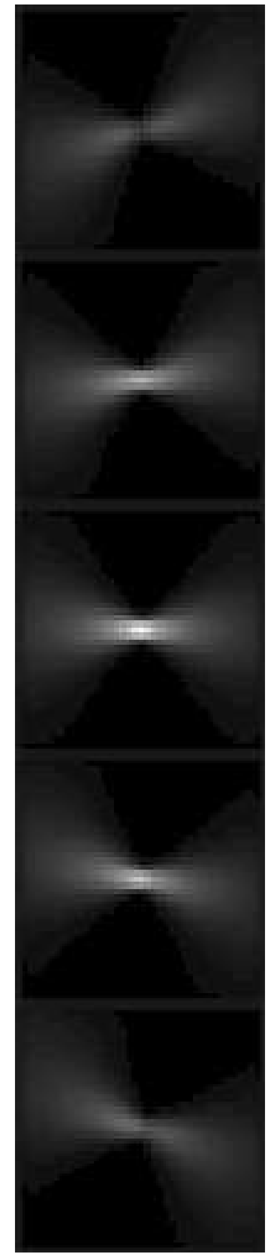
$\theta = 45^\circ$

$\theta = 22.5^\circ$

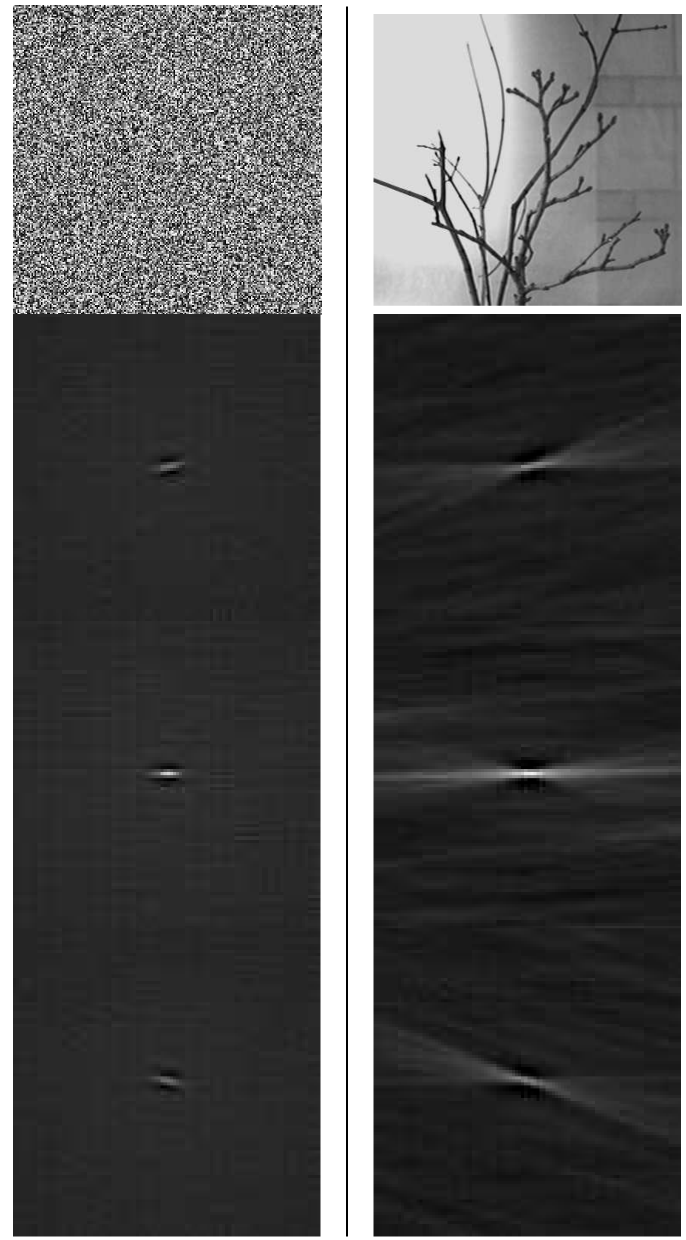
$\theta = 0^\circ$

$\theta = -22.5^\circ$

$\theta = -45^\circ$



Edge Correlations Observed in Images



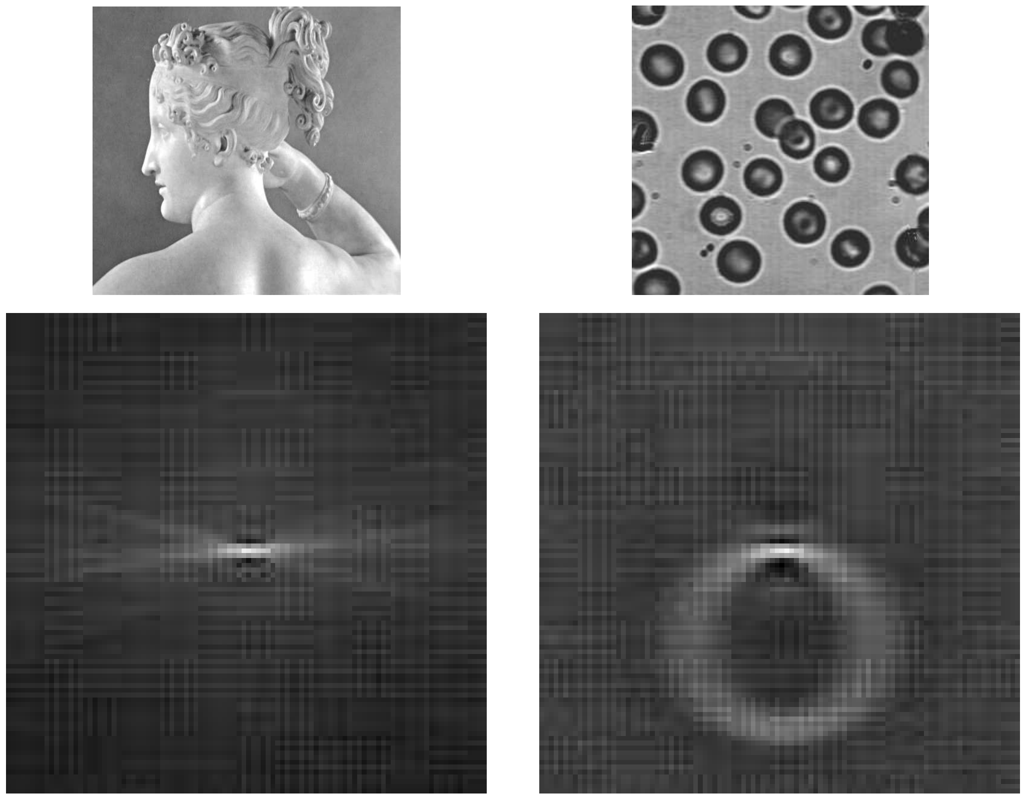
Original image

$\theta = 22.5^\circ$

$\theta = 0^\circ$

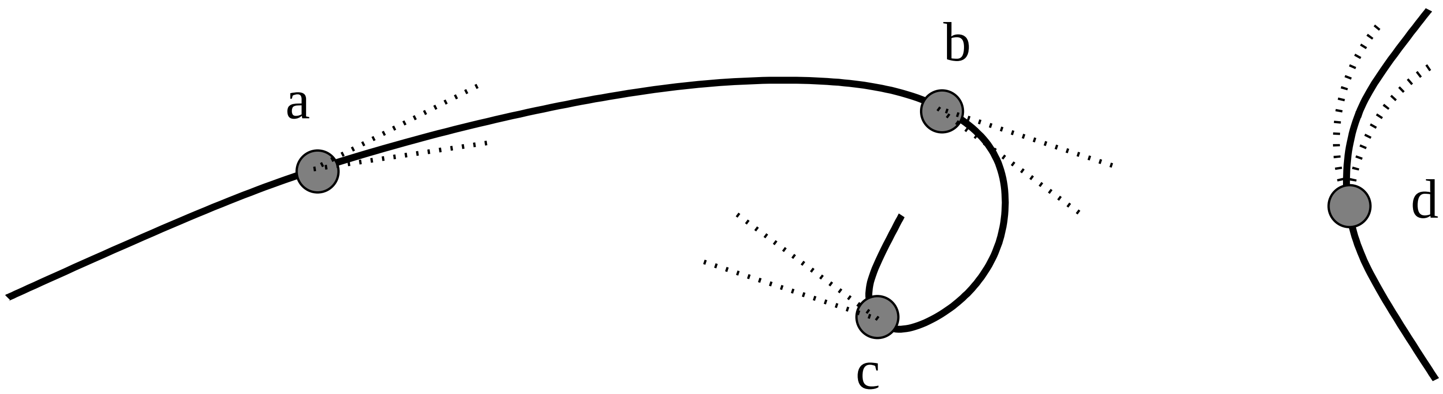
$\theta = -22.5^\circ$

The Need for Curvature



Edge correlations (integrated over θ)

The Benefit of Curvature



Curvature "tunes" search window

A Markov Process with Curvature

Lift *with curvature*: $t \mapsto R_t = (x, y, \theta, \kappa)(t)$

Brownian motion in *curvature*:

$$\dot{x} = \cos\theta \quad \dot{y} = \sin\theta \quad \dot{\theta} = \kappa \quad \dot{\kappa} = \text{noise}$$

Most probable curve minimizes: $\alpha \int \dot{\kappa}^2 + \beta \int dt \quad \leftrightarrow \quad$ Euler spiral

Fokker-Plank diffusion: $\frac{\partial p}{\partial t} = Qp, \quad$ where $Q := \frac{\sigma^2}{2}\frac{\partial^2}{\partial \kappa^2} - \cos\theta \frac{\partial}{\partial x} - \sin\theta \frac{\partial}{\partial y} - \kappa \frac{\partial}{\partial \theta} - \alpha$

Q: "killed" Markov process "generator"

A Markov Process with Curvature

Lift *with curvature*: $t \mapsto R_t = (x, y, \theta, \kappa)(t)$

Brownian motion in *curvature*:

$$\dot{x} = \cos\theta \quad \dot{y} = \sin\theta \quad \dot{\theta} = \kappa \quad \dot{\kappa} = \text{noise}$$

Most probable curve minimizes: $\alpha \int \dot{\kappa}^2 + \beta \int dt \quad \leftrightarrow \quad$ Euler spiral

Fokker-Plank diffusion: $\frac{\partial p}{\partial t} = Qp, \quad$ where $Q := \frac{\sigma^2}{2}\frac{\partial^2}{\partial \kappa^2} - \cos\theta \frac{\partial}{\partial x} - \sin\theta \frac{\partial}{\partial y} - \kappa \frac{\partial}{\partial \theta} - \alpha$

Q: "killed" Markov process "generator"

Compare to direction process [Mumford]:

$$\dot{x} = \cos\theta \quad \dot{y} = \sin\theta \quad \dot{\theta} = \text{noise}$$

Most probable curve minimizes: $\alpha \int \kappa^2 + \beta \int dt \quad \leftrightarrow \quad$ Elastica

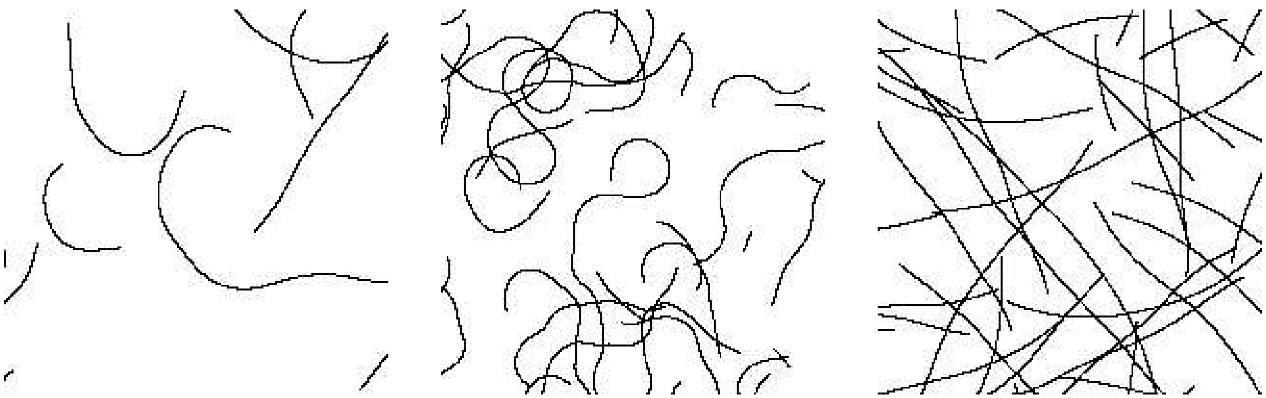
Fokker-Plank diffusion: $\frac{\partial p}{\partial t} = Qp, \quad$ where $Q := \frac{\sigma^2}{2}\frac{\partial^2}{\partial \theta^2} - \cos\theta \frac{\partial}{\partial x} - \sin\theta \frac{\partial}{\partial y} - \alpha$

Sketches Compared

CIRF Samples With Direction Only

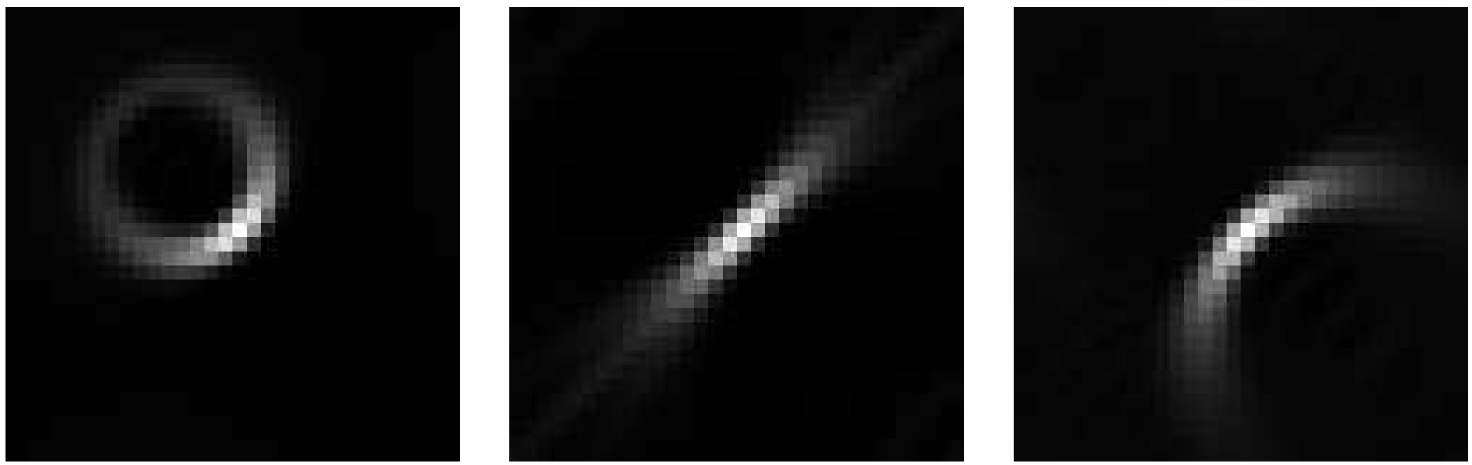
Sketches Compared

CIRF Samples With Direction Only



CIRF Samples **Including Curvature**

Curve Indicator Random Field Covariance with Curvature:



κ_0: 0.2 0 -0.1

Moment Generating Functional

For (multi-curve) curve indicator random field U:

$$\mathbb{E}\exp(c, U) = \exp(\mu, \bar{N}(G(c) - G)\nu),$$

where:

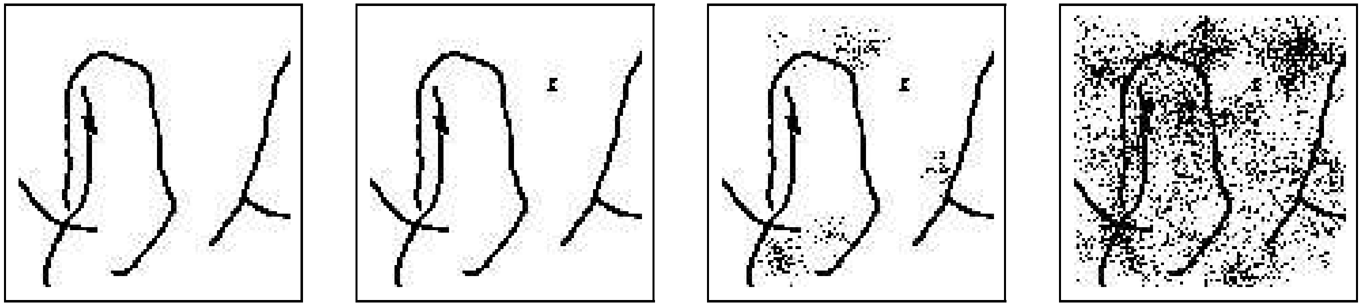
$$\begin{aligned}
Q &= \text{ killed Markov process generator (e.g., direction or curvature process)} \\
G &= -Q^{-1} = \text{ Green's function} \\
G(c) &= -(Q + \operatorname{diag} c)^{-1} = \text{ Green's function biased by } c \\
\mu &= \text{ initial weighting} \\
\nu &= \text{ final weighting} \\
\bar{N} &= \text{ average number of curves}
\end{aligned}$$

Observe: Linear system.

Minimum Mean-Square Estimation of the CIRF

Bayes estimate: $\tilde{u}(m) = \arg\min_u \mathbb{E}[\text{loss}(U, u)|m]$

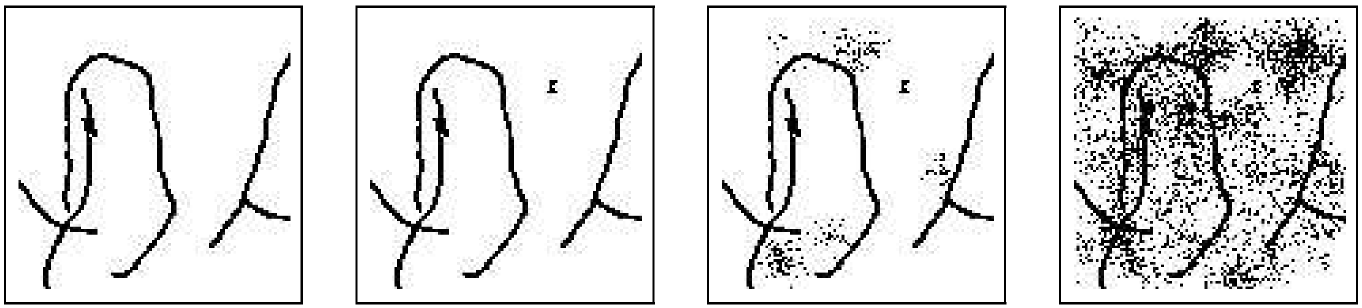
U = CIRF, m = measurements, loss = mean-square error.



Minimum Mean-Square Estimation of the CIRF

Bayes estimate: $\tilde{u}(m) = \arg\min_u \mathbb{E}[\text{loss}(U,u)|m]$

U = CIRF, m = measurements, loss = mean-square error.

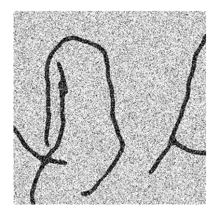


Goal:

Filter Output = Minimum mean square error estimate (MMSE) of U
= Posterior mean of CIRF U (given measurements m)

Likelihood:

Assume Gaussian white noise, blur B, Poisson number of curves.



High-noise MMSE CIRF Volterra Filters

Assume no blur and white Gaussian noise, variance $\sigma_N^2 = \epsilon^{-1}$.
High-noise limit: Take Taylor expansion of log normalizing constant of posterior around $\epsilon = 0$. (ζ = constant)

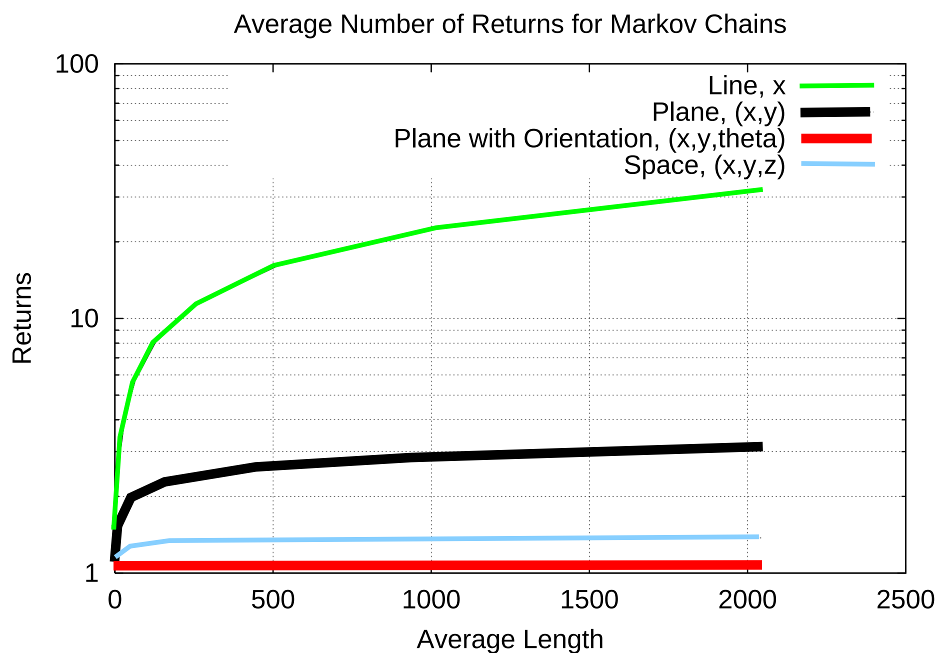
Low contour density η assumption.

$$\tilde{u}^{(1)} = \eta\{1 - 2\epsilon\zeta + \epsilon(Gm + G^*m)\}$$

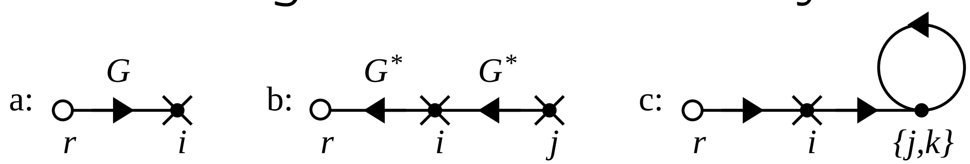
$$\begin{aligned} \tilde{u}^{(2)} = \eta\{&1 - 2\epsilon\zeta + 3\epsilon^2\zeta^2 + \epsilon(1 - 2\epsilon\zeta)(Gm + G^*m) \\ &+\epsilon^2(G \text{ diag}\, m\, Gm + Gm \odot G^*m + G^* \text{ diag}\, m\, G^*m)\} \end{aligned}$$

$$\begin{aligned} \tilde{u}^{(3)} = \eta\{&1 - 2\epsilon\zeta + 3\epsilon^2\zeta^2 - 4\epsilon^3\zeta^3 \\ &+\epsilon(1 - 2\epsilon\zeta + 3\epsilon^2\zeta^2)(Gm + G^*m) \\ &+\epsilon^2(1 - 2\epsilon\zeta)(G \text{ diag}\, m\, Gm + Gm \odot G^*m + G^* \text{ diag}\, m\, G^*m) \\ &+\epsilon^3(G \text{ diag}\, m\, G \text{ diag}\, m\, Gm + G \text{ diag}\, m\, Gm \odot G^*m \\ &\quad +Gm \odot G^* \text{ diag}\, m\, G^*m + G^* \text{ diag}\, m\, G^* \text{ diag}\, m\, G^*m)\} \end{aligned}$$

Self-Avoiding Curves

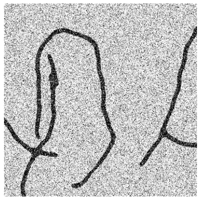


- Derivation based on diagrams similar to Feynman diagrams



- Many diagrams produce negligible terms due to self-avoidance

MMSE CIRF Filtering via Nonlinear PDEs



Assume Gaussian white noise, blur B, Poisson number of curves.

Goal:

Filter Output $=$ Posterior mean of CIRF U (given measurements M)

Exact prior + approximate likelihood

$\rightarrow$ biased CIRF approximation of posterior mean:

$$(Q + \text{diag}\, d)\, f = \text{const} \qquad \text{Forward PDE}$$

$$(Q^* + \text{diag}\, d)\, b = \text{const} \qquad \text{Backward PDE}$$

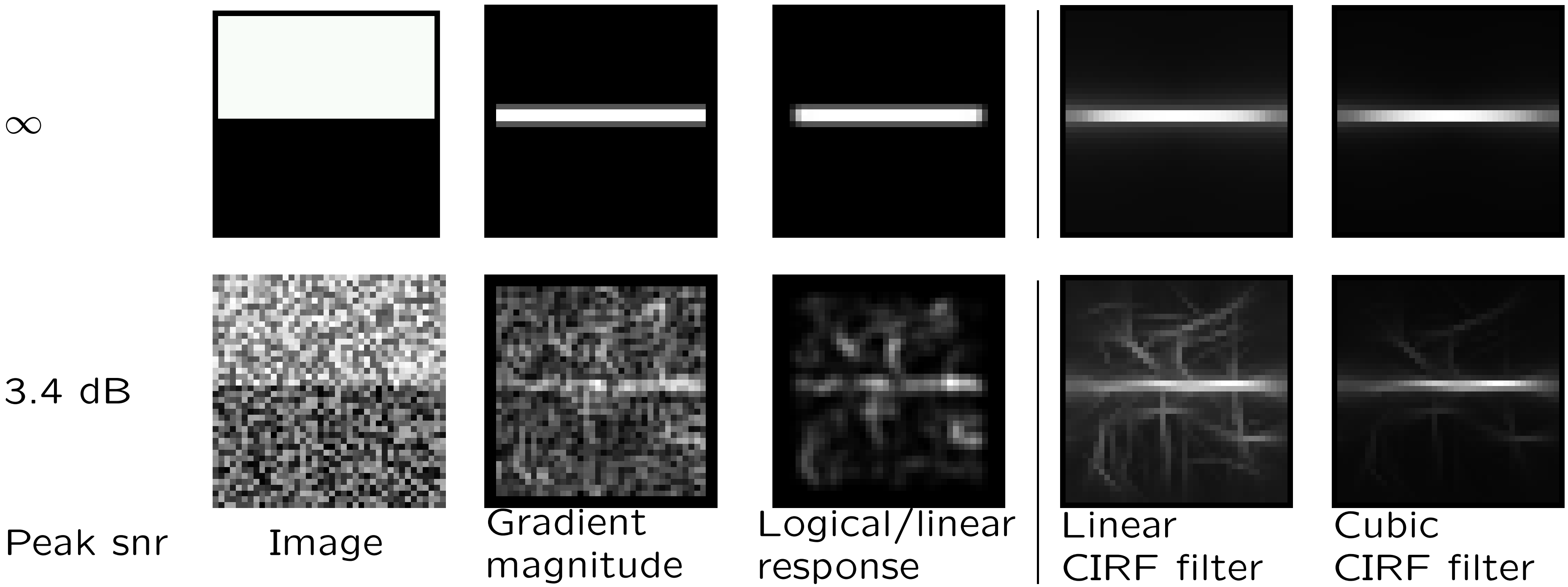
$$d = \epsilon B^*(M - B(f \odot b))$$

$$\text{Filter Output}_i = f_i\, b_i \approx \mathbb{E}_M U_i$$

$$Q = \text{killed Markov process generator}$$

– Reaction-diffusion-convection equation

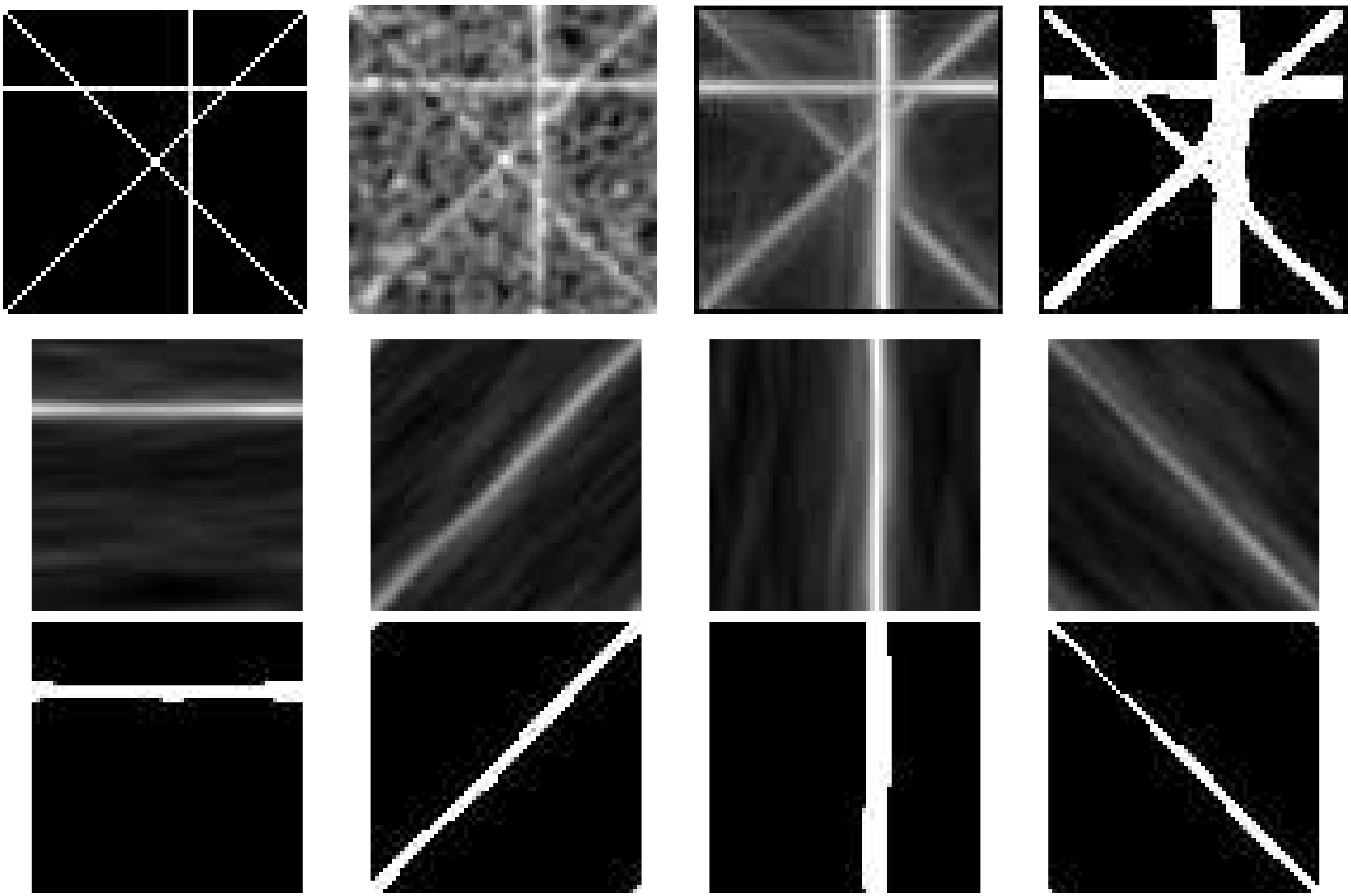
Effect of Filters in (x, y, θ)



Nonlinear CIRF PDE filter: Noise Result

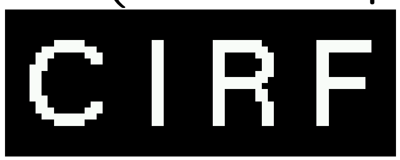
(Result is function of (x, y, θ). Integrated over θ for display.)

Pick Up Sticks

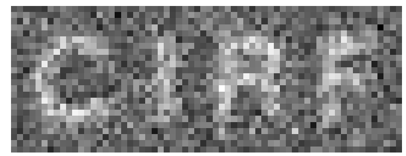


0°   45°   90°   135°

Original (No corruption)

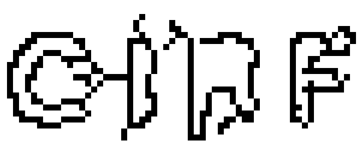


With Blur and Noise

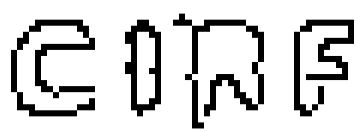


Thresholding of Filter Output

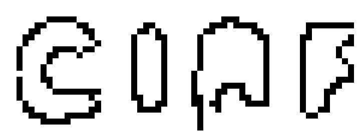
Canny:



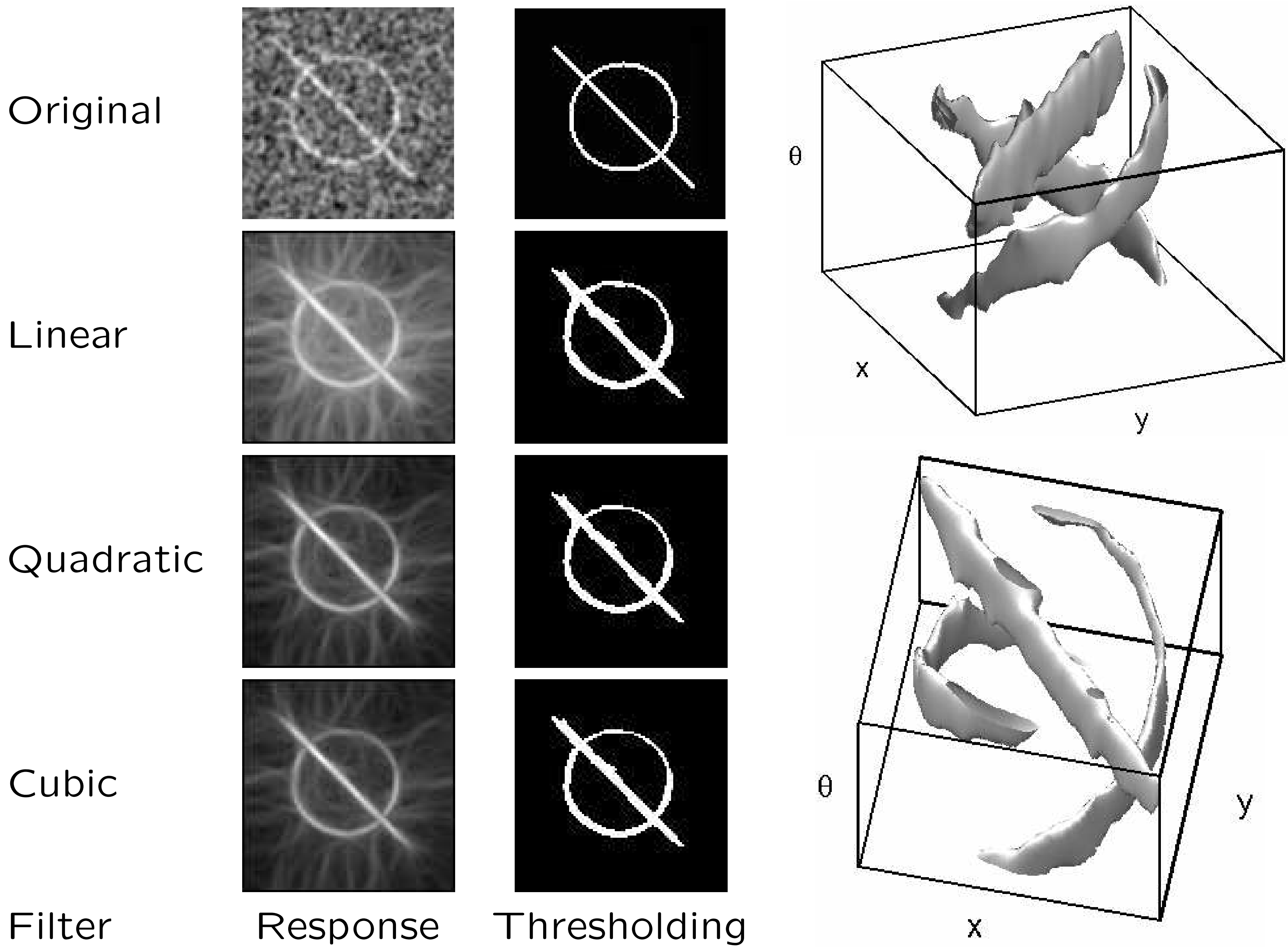
$\sigma = 1$



$\sigma = 1.5$



$\sigma = 2$


Original
Linear
Quadratic
Cubic
Filter
Response
Thresholding
θ
x
y
θ
y
x

Finding a Ship's Wake

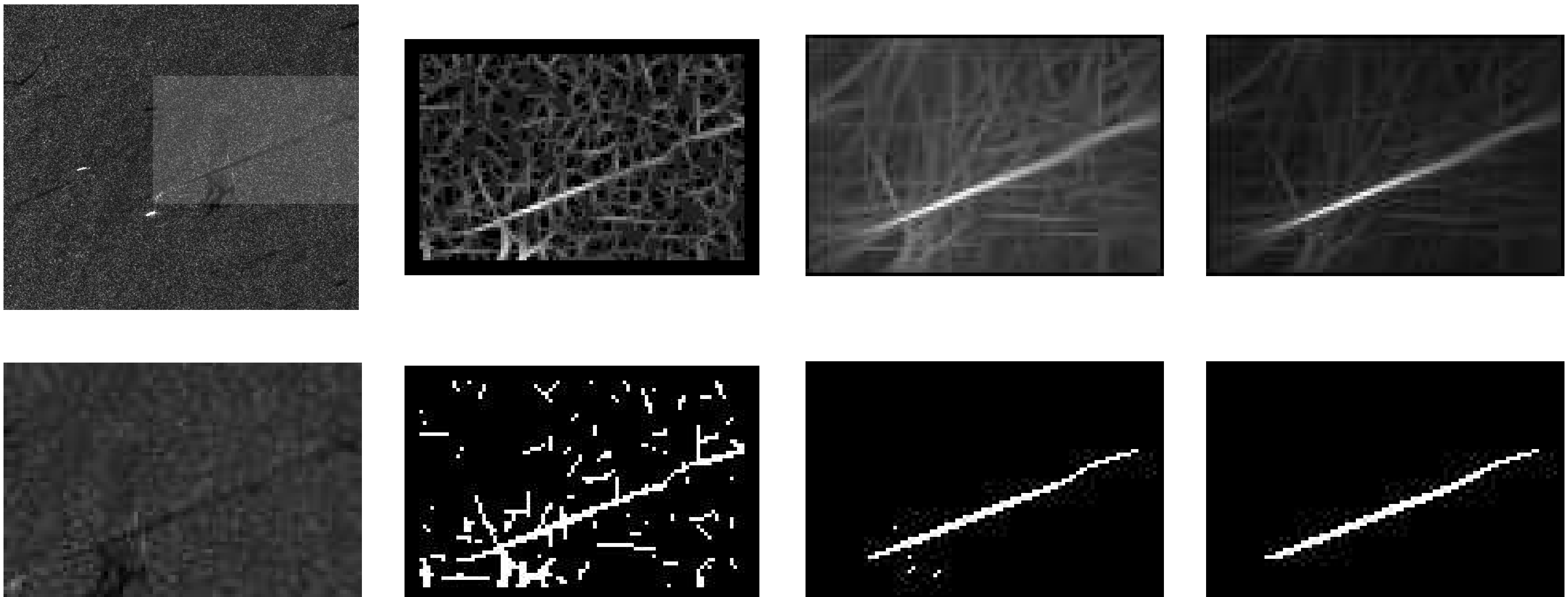
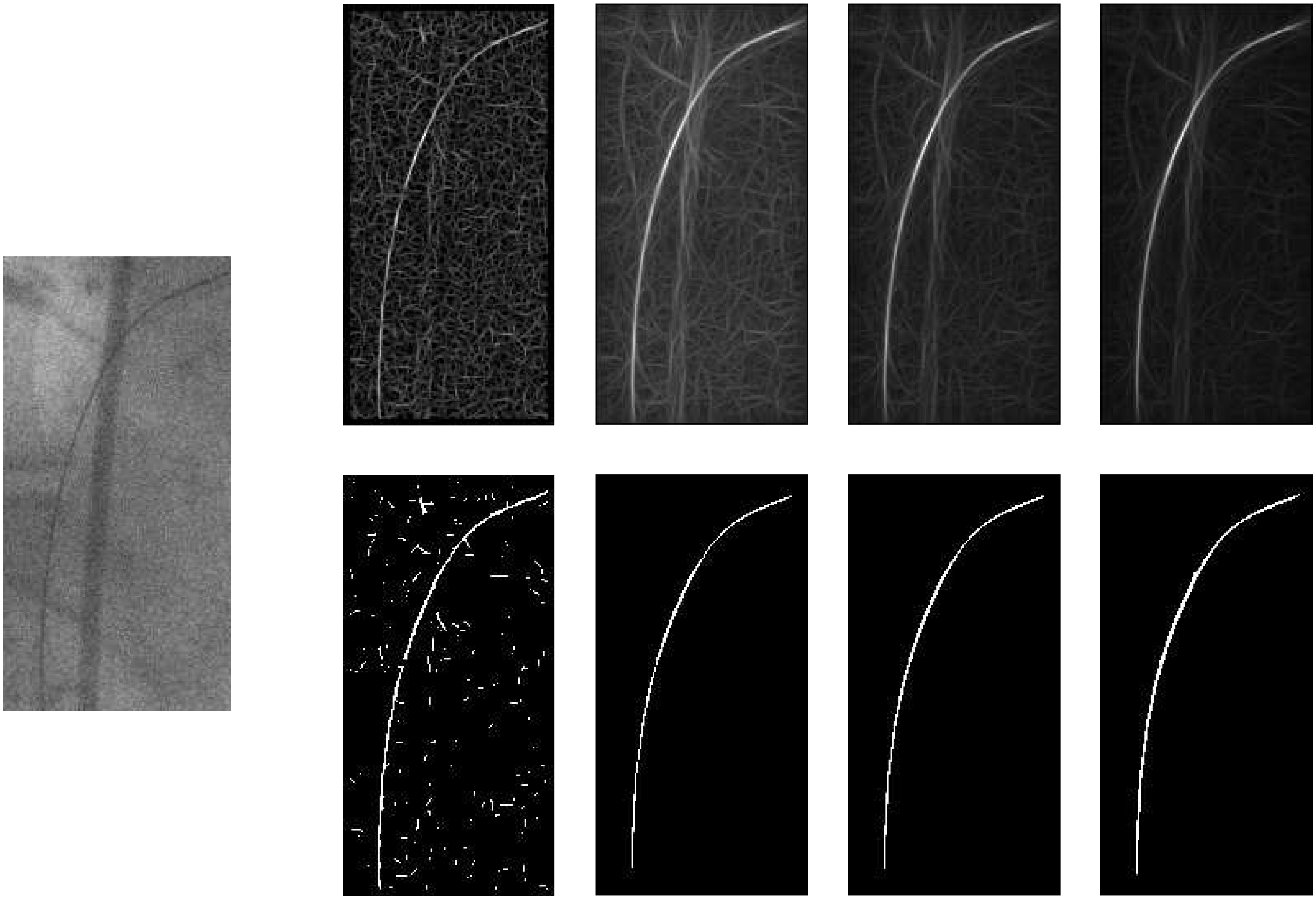


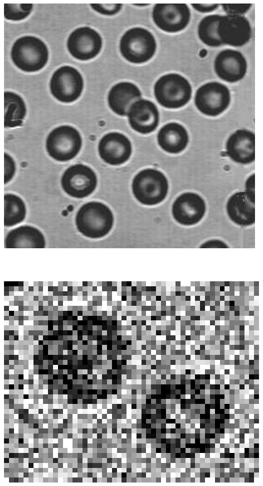
Image | Local Responses | Linear | Cubic

Finding a Surgical Guide Wire

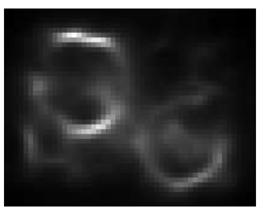


Local Responses | Linear | Quadratic | Cubic

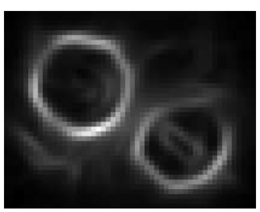
Filtering with Curvature



Original

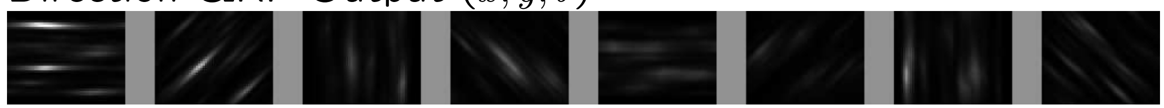


Direction CIRF

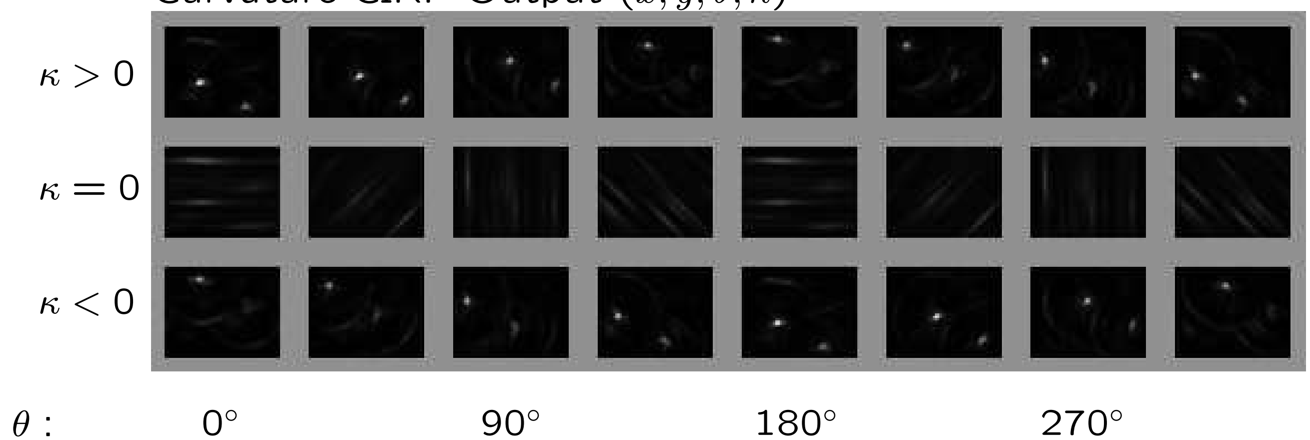


Curvature CIRF

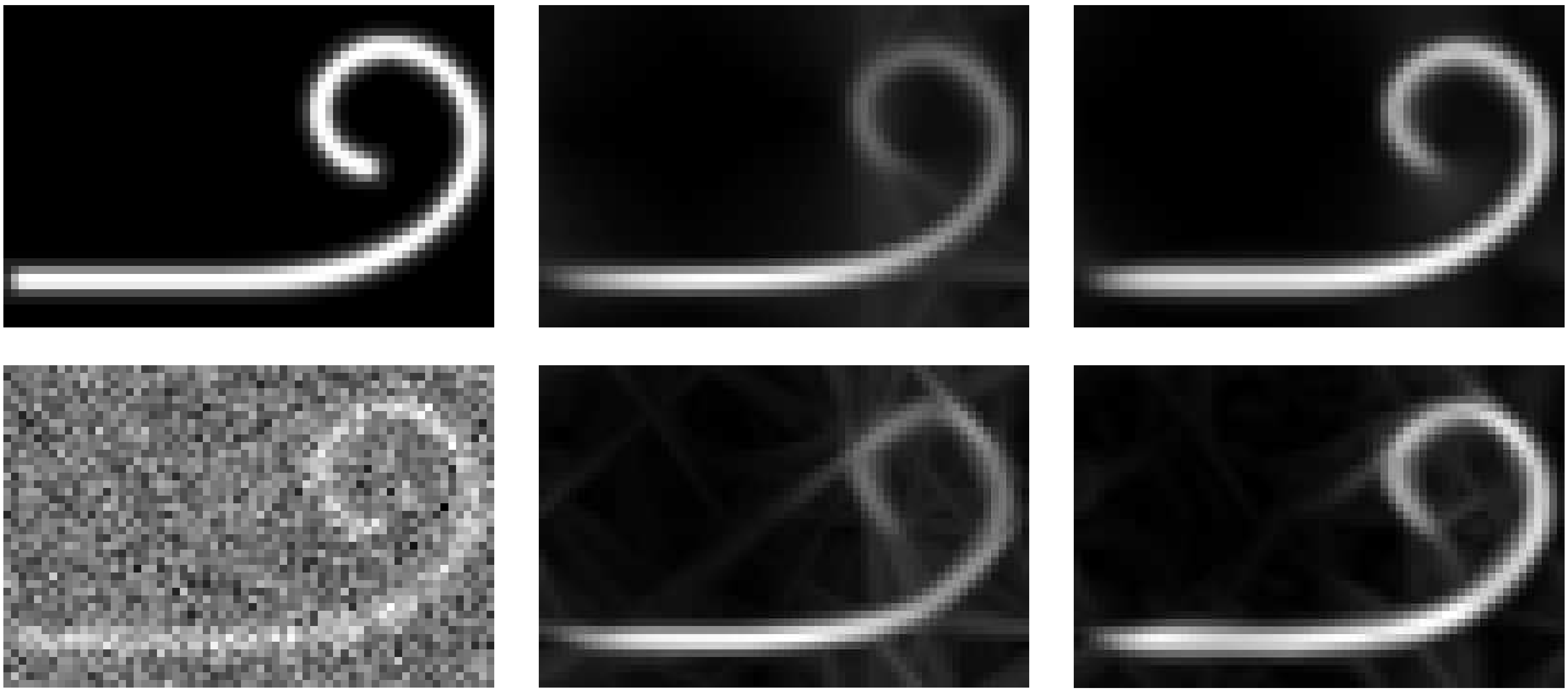
Direction CIRF Output (x, y, θ)



Curvature CIRF Output (x, y, θ, κ)

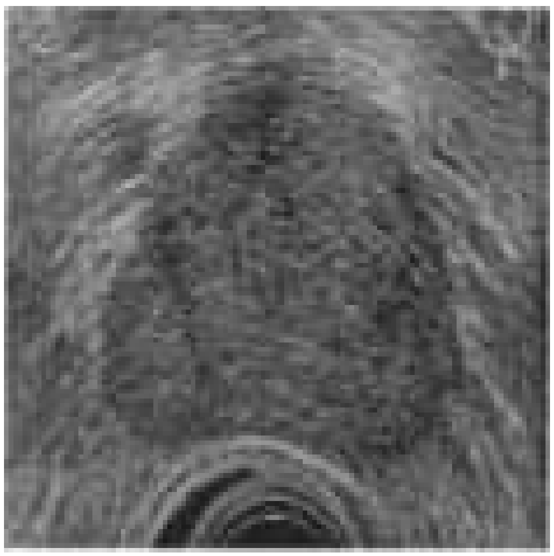


Filtering an Euler Spiral

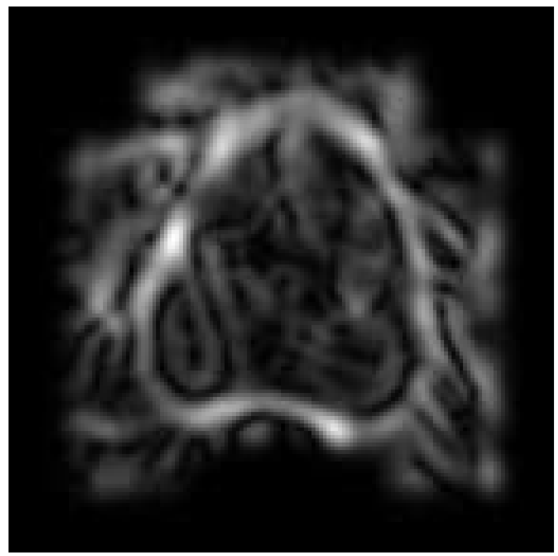


Original | Direction CIRF Filtered | Curvature CIRF Filtered

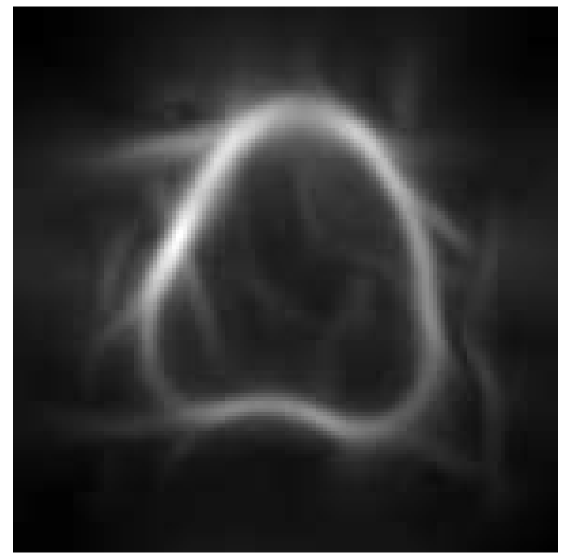
Prostate Enhancement



Original



Edges before



Edges after
cubic CIRF filter

Conclusions

- Differential Geometry: Stochastic Model of Contour Curvature
- Inference: Posterior Mean Filter using nonlinear PDEs
- Curve Indicator Random Field as:
  - Sketch (Ideal Edge Map)
  - Abstraction for Eliminating Curve Parameterization